A Recurrent Neural Network-based Surrogate Model for History-Dependent Multi-scale Simulations

L. Wu , L. Noels

University of Liege, IMDEA Materials, JKU

Computational & Multiscale Mechanics of Materials

http://www.ltas-cm3.ulg.ac.be

SimTech Seminar on Model Reduction and Data Techniques for Surrogate Modelling, University of Stuttgart

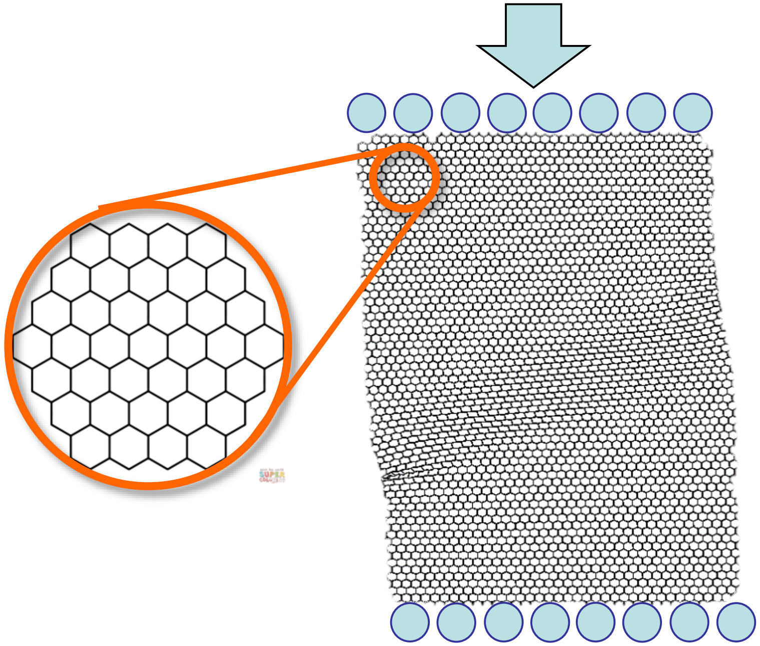
04 November 2021

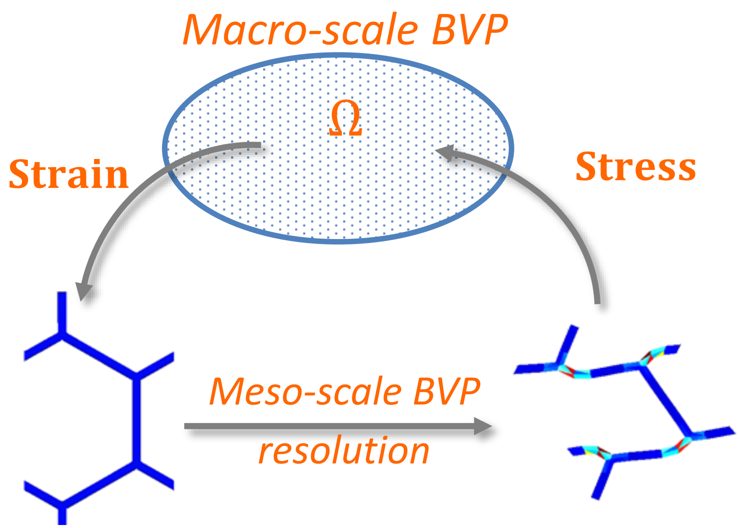
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 862015

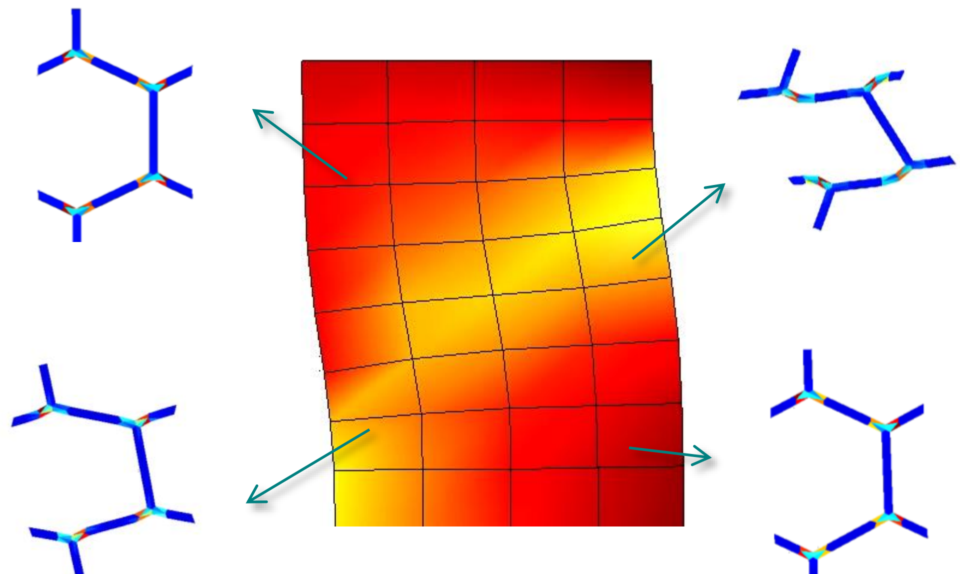
MOAMMM

www.moammm.eu

- Computational homogenisation (FE2)
  - Heterogeneous structures
    - Micro-scale: cell, grains, inclusions…
    - Macro-scale: seen as a continuum
  - Direct numerical simulations
    - Time consuming
  - Idea: use multi-scale strategy



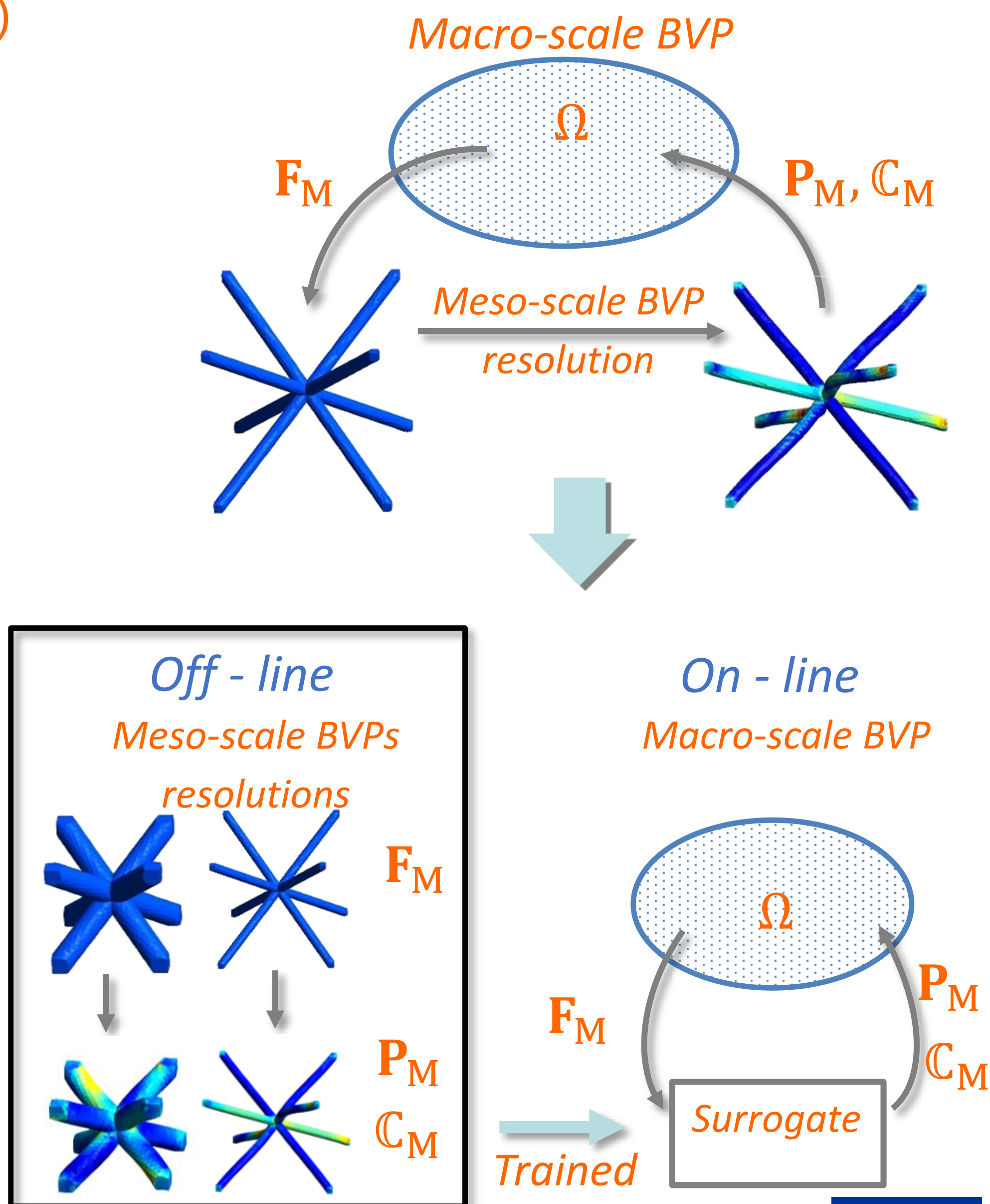




- Computational homogenisation (FE2)
  - Non-linear simulations
    - Iterations at macro-scale BVP
    - Sub-iterations at meso-scale BVP

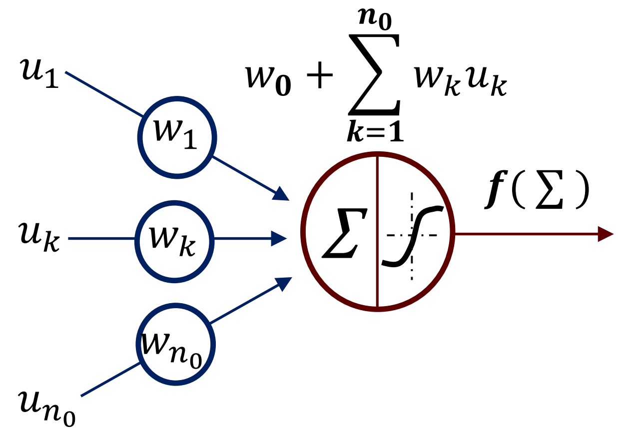
Unaffordable

  - Introduction of data-driven approach
  - Use of surrogate models
    - Train a surrogate model (off-line)
      - Requires extensive data
      - Obtained from RVE simulations
    - Use the trained surrogate model during analyses (on-line)
      - Speed-up of several orders

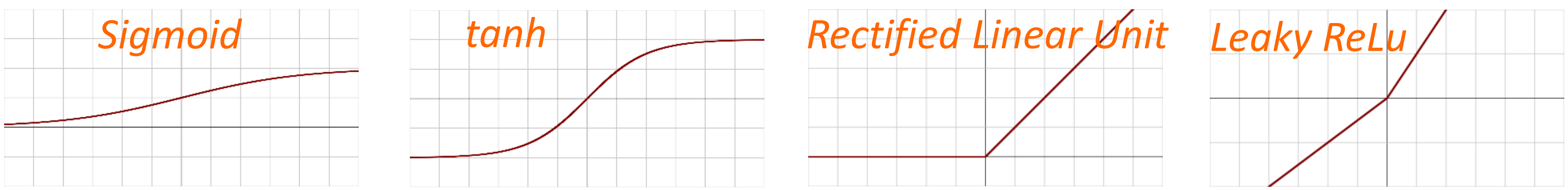


Artificial Neural Network

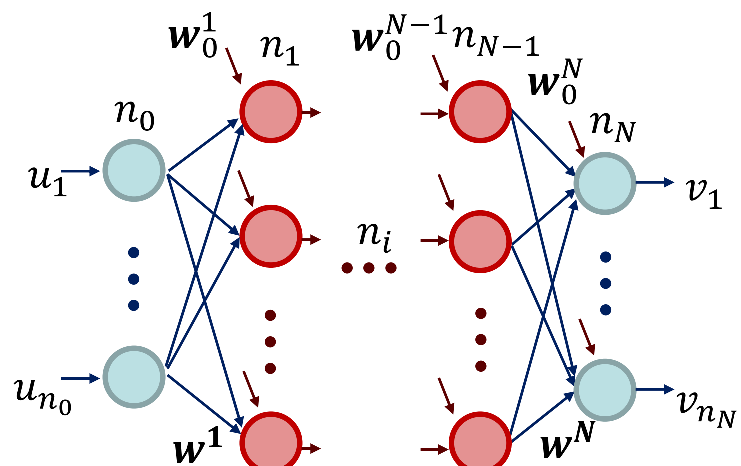
- Definition of the surrogate model
  - Artificial neuron
    - Non-linear function on n_0 inputs u_k
    - Requires evaluation of weights w_k
    - Requires definition of activation function f



  - Activation functions f



  - Feed-Forward Neuron Network
    - Simplest architecture
    - Layers of neurons
      - Input layer
      - $N-1$ hidden layers
      - Output layers
    - Mapping $\Re^{n_0} \rightarrow \Re^{n_N}: \boldsymbol{v} = \boldsymbol{g}(\boldsymbol{u})$



Artificial Neural Network

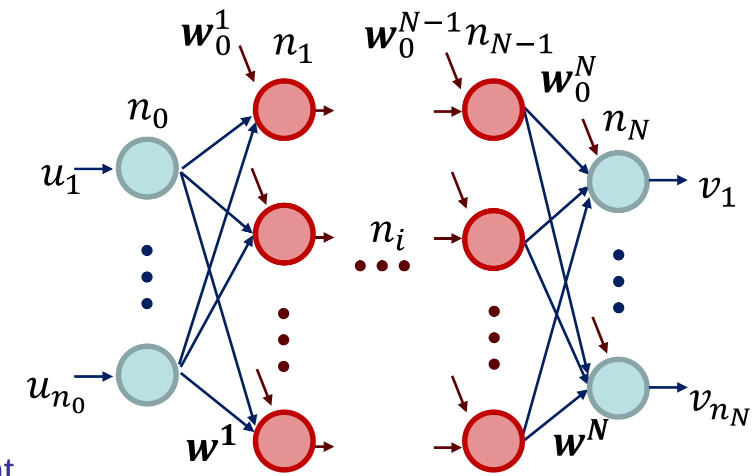
- Training
  - Evaluate
    - The weights w_{kj}^{i}, $k = 1..n_{i-1}, j = 1..n_i$
    - The bias w_0^i
    - Minimise error prediction $\boldsymbol{v}$ vs. real $\boldsymbol{v}^{(p)}$

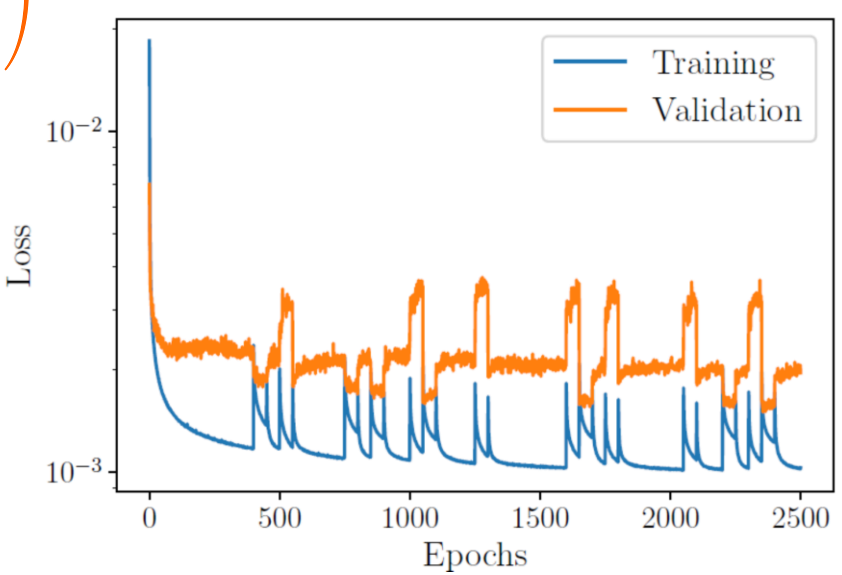
$$L_{\text{MSE}}\,(\mathbf{W}) = \frac{1}{n}\sum_{i}^{n}\left\|\boldsymbol{v}_i(\mathbf{W}) - \boldsymbol{v}_i^{(p)}\right\|^2$$

    - Requires an optimizer: Stochastic Gradient Descent

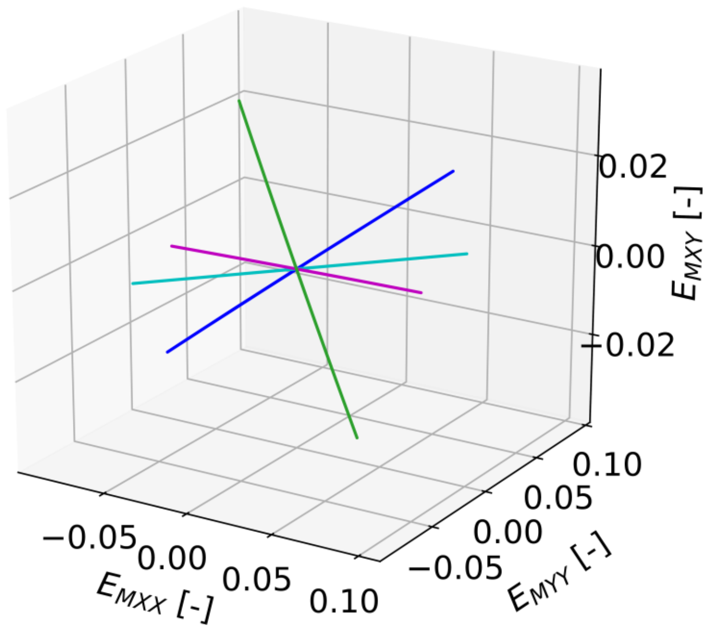
$$\mathbf{\Delta W} = -\mathcal{F}\left(\frac{\partial L_i(\mathbf{W})}{\partial \mathbf{W}}, \quad \left(\frac{\partial L_i(\mathbf{W})}{\partial \mathbf{W}}\right)^2, \text{batch size}, \ldots\right)$$

  - Training data
    - Input $\boldsymbol{u}^{(p)}$ & Output $\boldsymbol{v}^{(p)}$
- Testing
  - Use new data
    - Input $\boldsymbol{u}^{(p)}$& Output $\boldsymbol{v}^{(p)}$
    - Verify prediction $\boldsymbol{v}$ vs. real $\boldsymbol{v}^{(p)}$

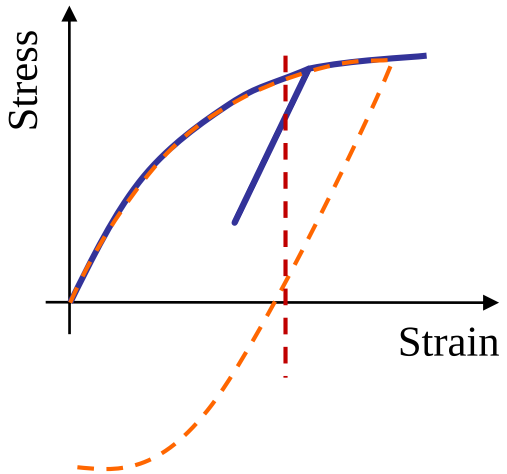




- Input / output definition
  - Input:
    - Strain (history): $\mathbf{F}_{\mathrm{M}}$
  - Output:
    - Stress (history): $\mathbf{P}_{\mathrm{M}}$

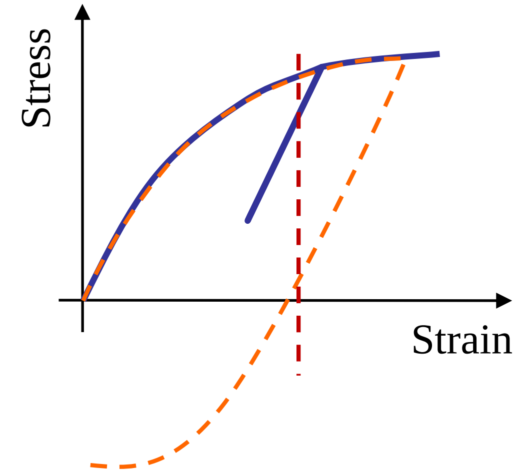


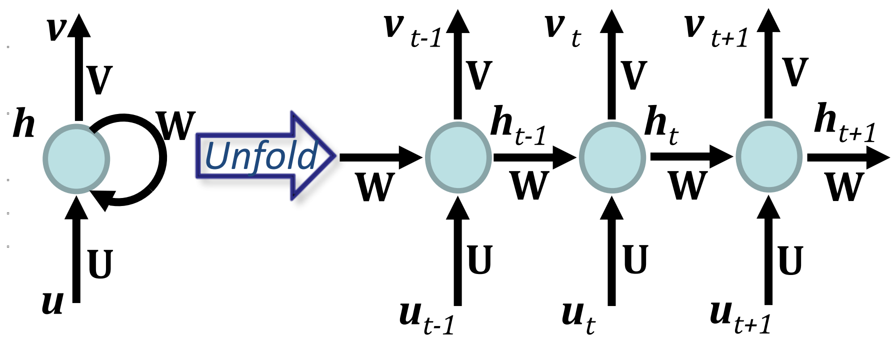
- Methodology
  - Address problem of history dependency
    - RVE without buckling
    - Elasto-plastic composite RVE



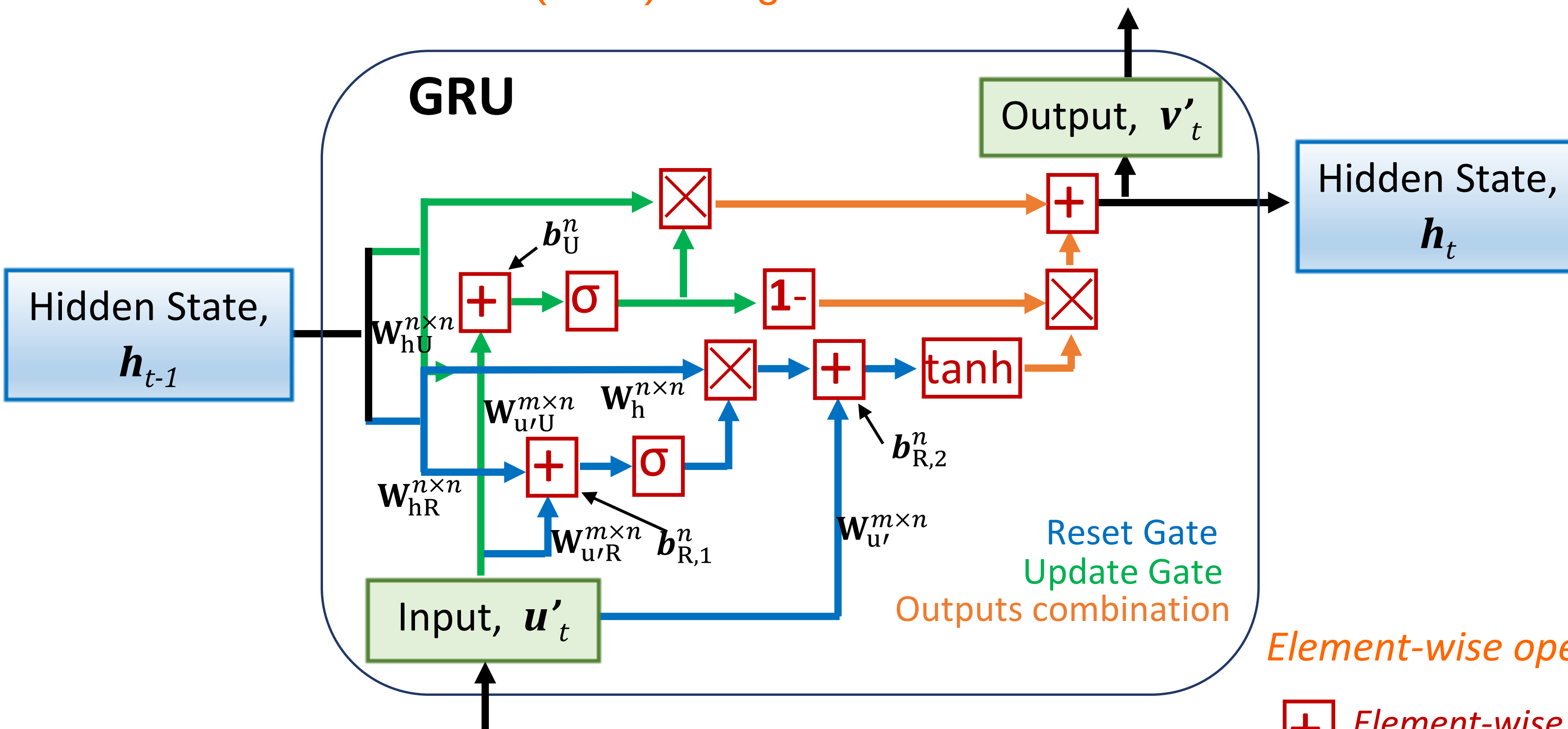
History dependency

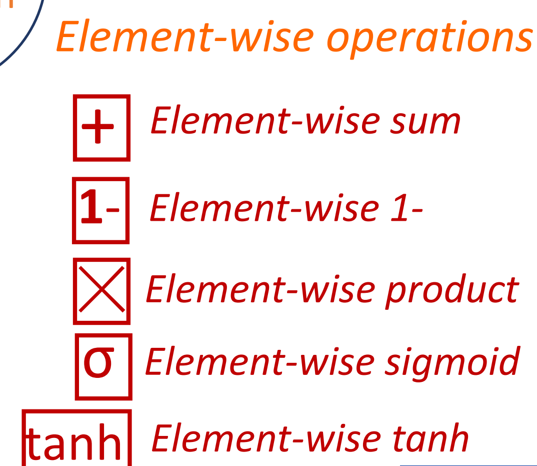
- Elasto-plastic material behaviour
  - No bijective strain-stress relation
    - Feed-forward NNW cannot be used
    - History should be accounted for
- Recurrent neural network
  - Allows a history dependent relation
    - Input $\boldsymbol{u}_t$
    - Output $\boldsymbol{v}_t = \boldsymbol{g}(\boldsymbol{u}_t, \boldsymbol{h}_{t-1})$
    - Internal variables $\boldsymbol{h}_t = \boldsymbol{g}(\boldsymbol{u}_t, \boldsymbol{h}_{t-1})$
  - Weights matrices $\mathbf{U}, \mathbf{W}, \mathbf{V}$
    - Trained using sequences
      - Inputs $\boldsymbol{u}_{t-n}^{(p)}, \ldots, \boldsymbol{u}_t^{(p)}$
      - Output $\boldsymbol{v}_{t-n}^{(p)}, \ldots, \boldsymbol{v}_t^{(p)}$

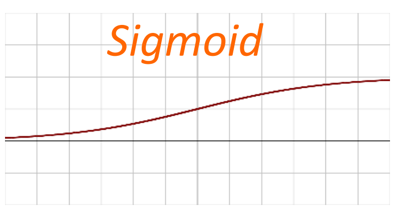


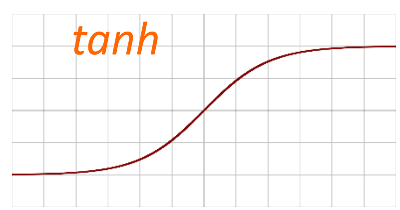


- Gated Recurrent Unit (GRU) at a glance

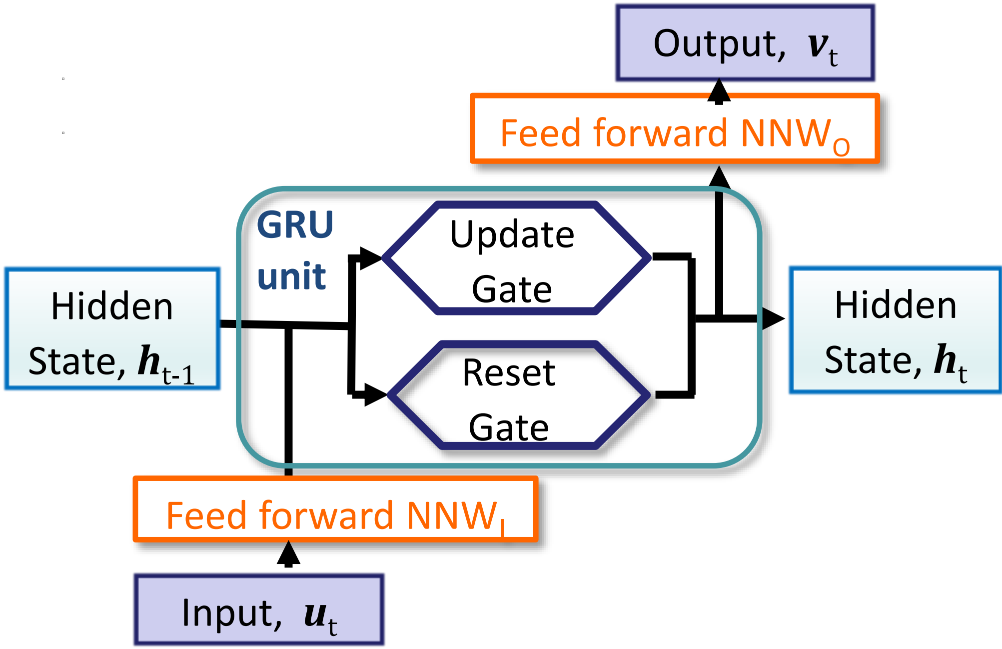


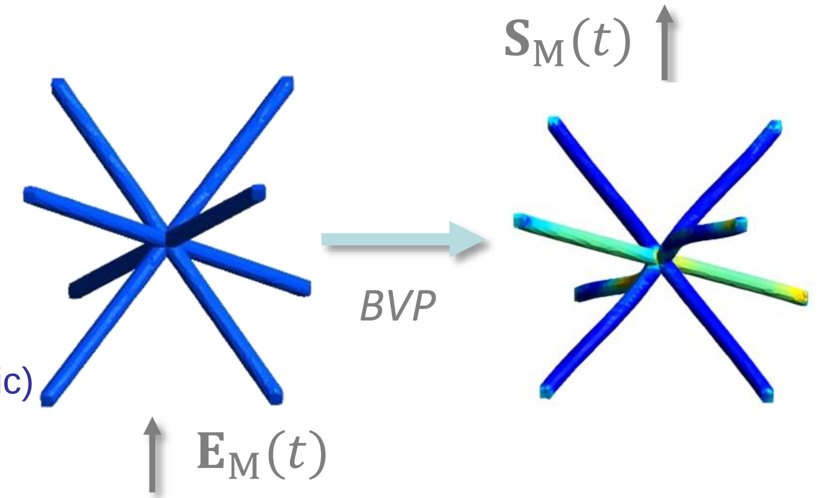




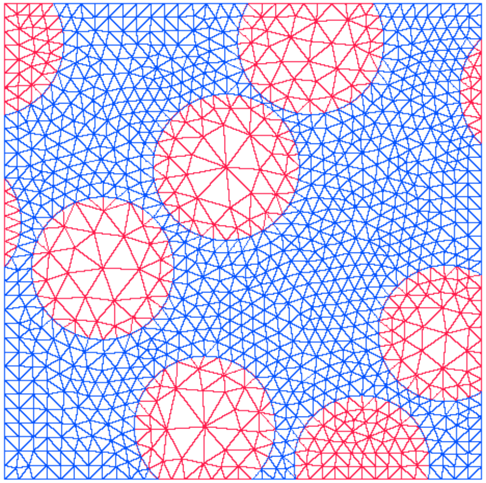


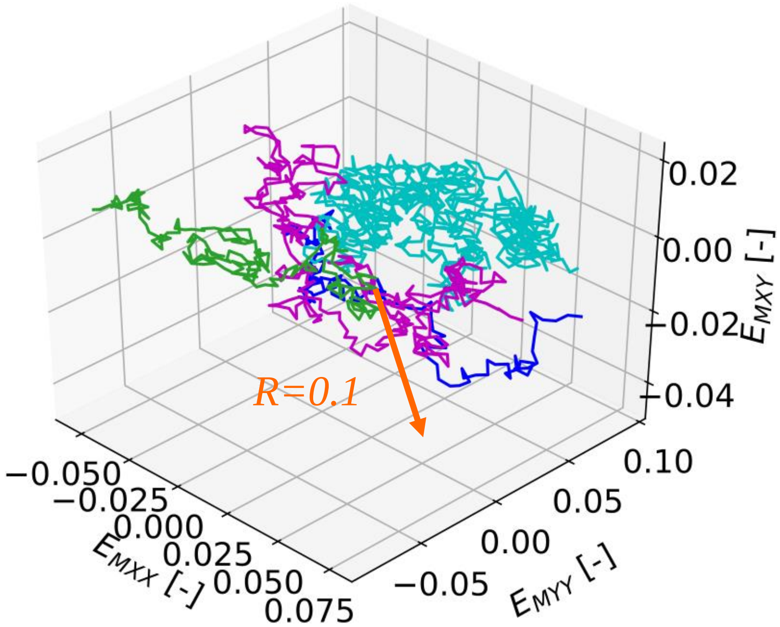
- Recurrent neural network design
  - 1 Gated Recurrent Unit (GRU)
    - Reset gate: select past information to be forgotten
    - Update gate: select past information to be passed along
    - Need to define number of hidden variables $\boldsymbol{h}_t$
  - 2 feed-forward NNWs
    - NNW_I to treat inputs $\boldsymbol{u}_t$
    - NNW_O to produce outputs $\boldsymbol{v}_t$
  - Input and Output
    - $\boldsymbol{u}_t$: homogenised GL strain $\mathbf{E}_\mathrm{M}$ (symmetric)
    - $\boldsymbol{v}_t$: homogenised 2nd PK stress $\mathbf{S}_\mathrm{M}$ (symmetric)

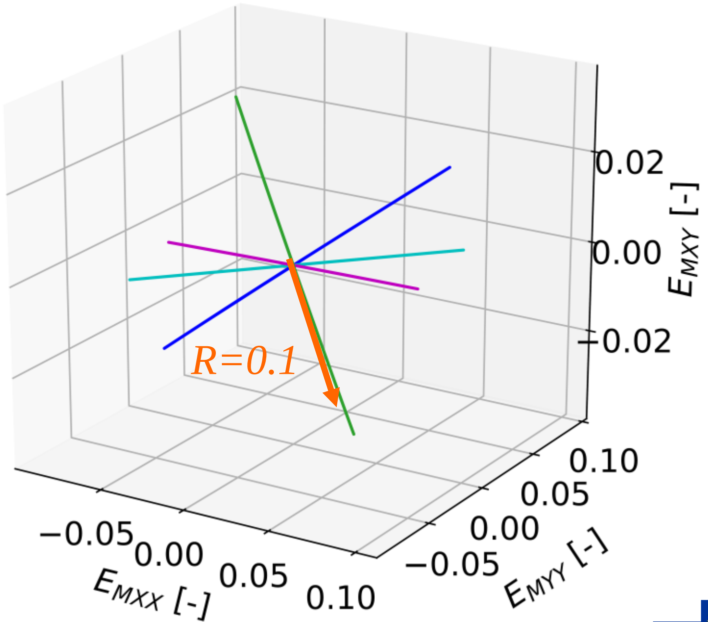




- Data generation
  - Elasto-plastic composite RVE
  - Training stage
    - Should cover full range of possible loading histories
    - Use random walking strategy (thousands)
    - Completed with random cyclic loading (tens)
    - Bounded by a sphere of 10% deformation

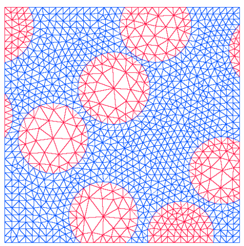


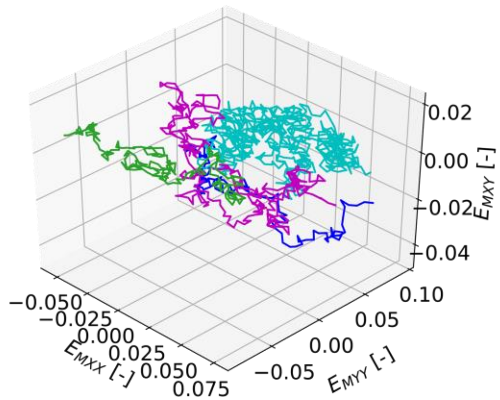


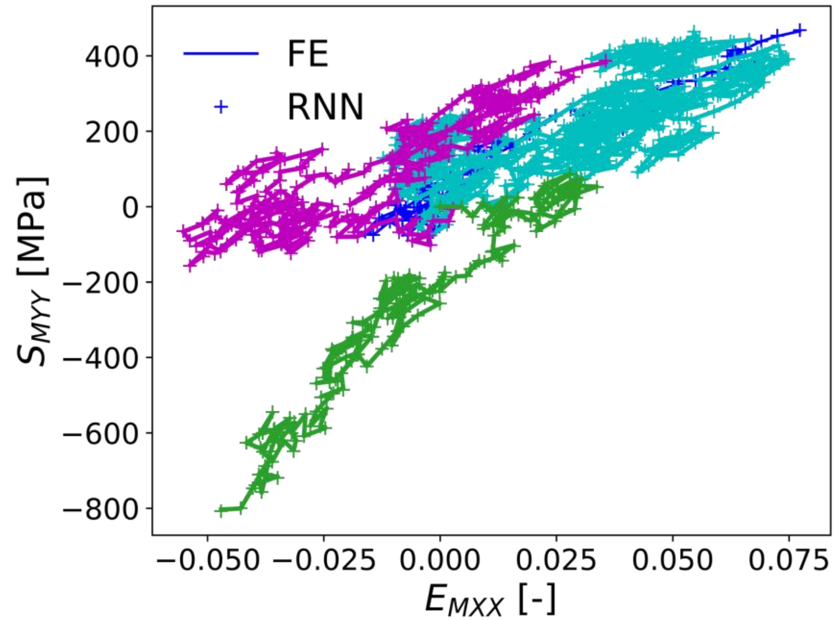


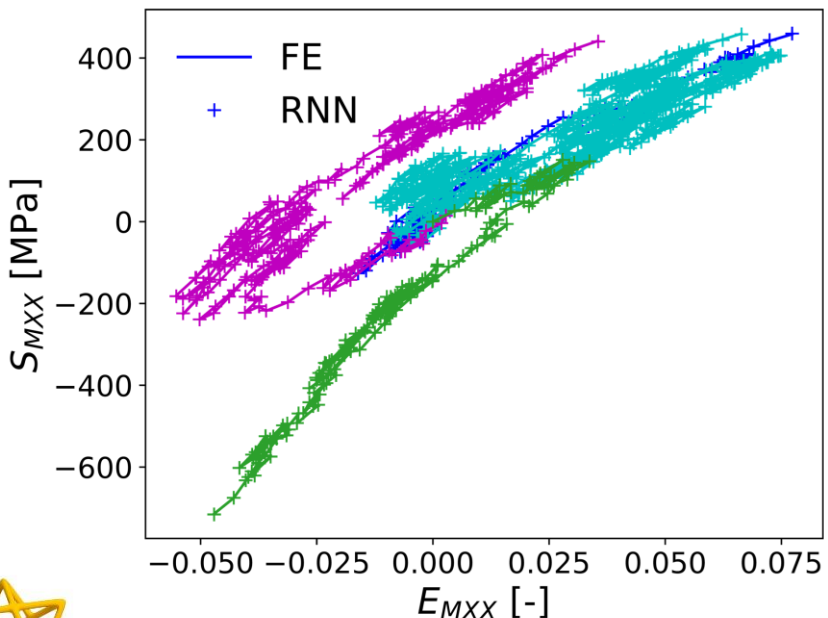
History dependency

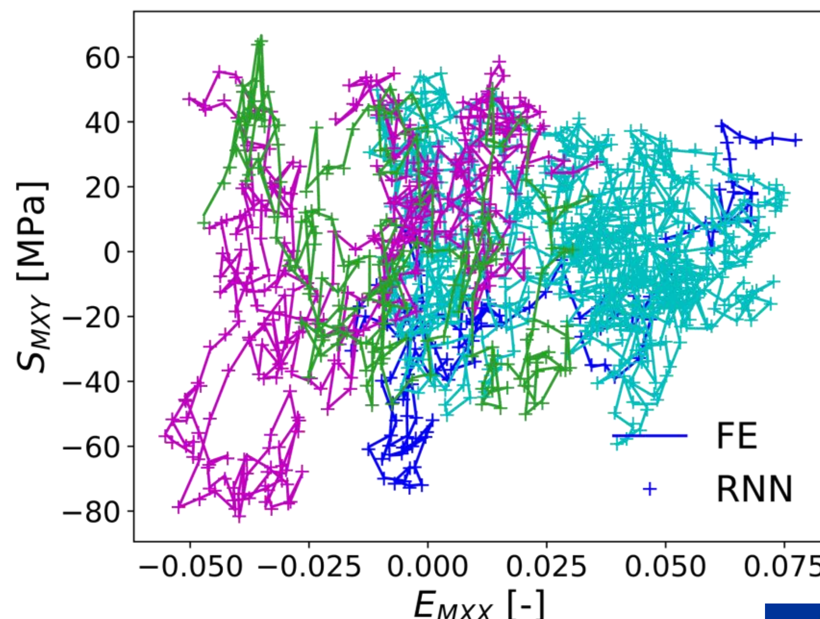
- Testing process (new data)
  - On random walk





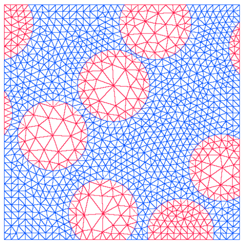


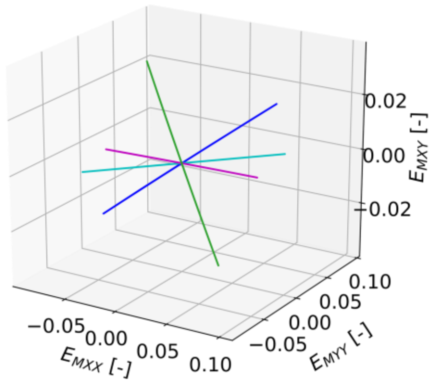


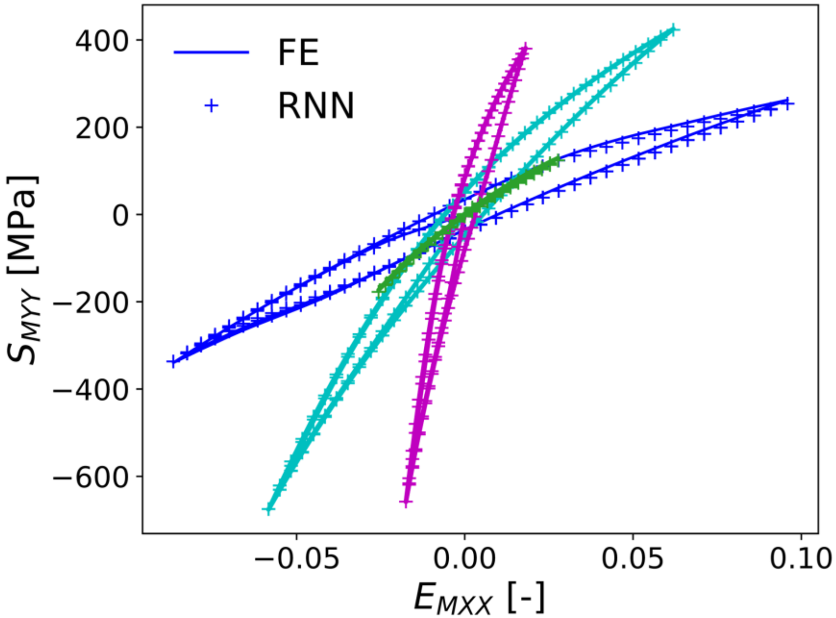


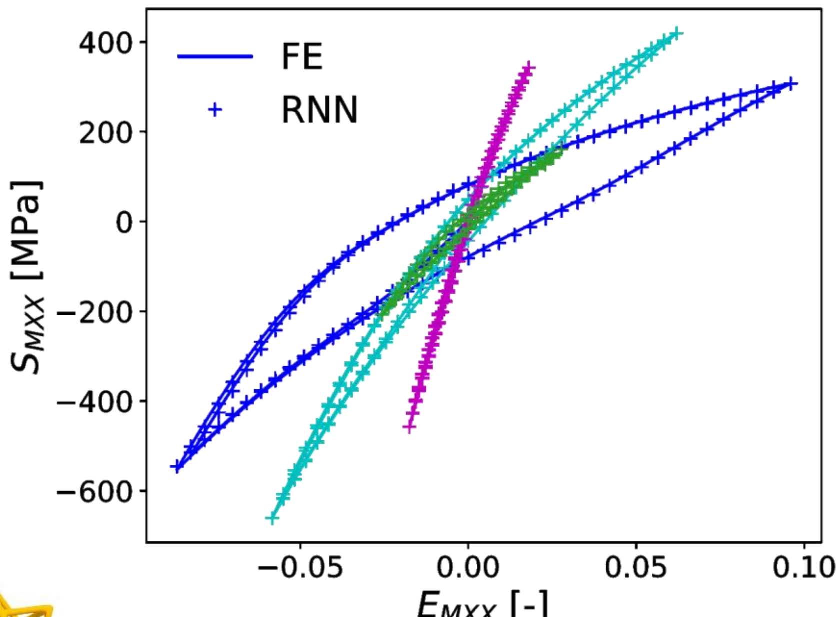
History dependency

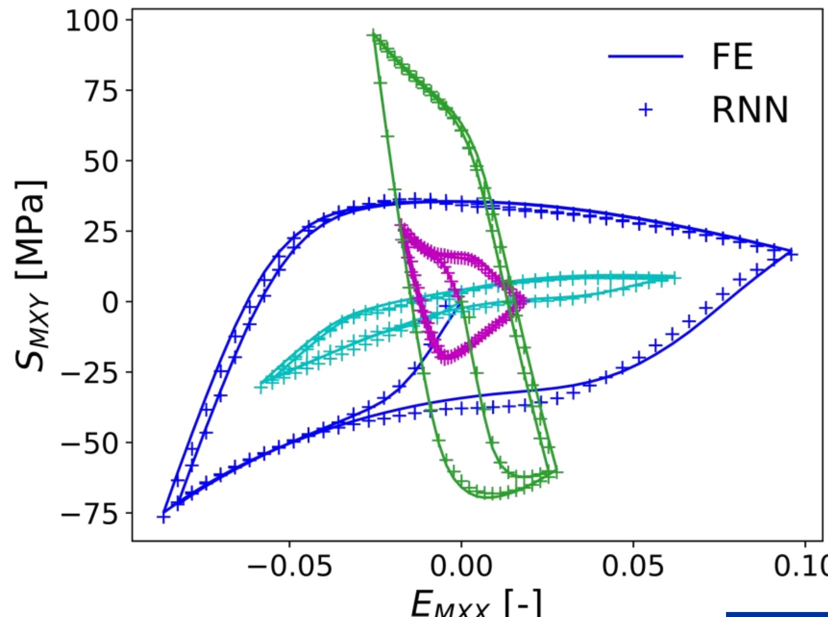
- Testing process (new data)
  - On cyclic loading







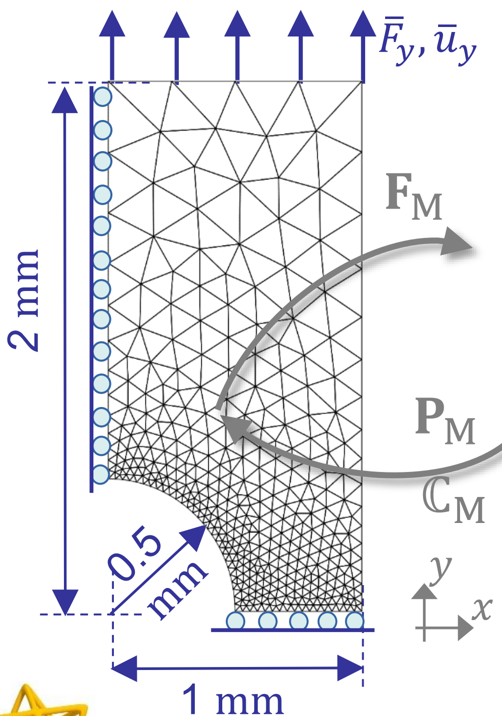


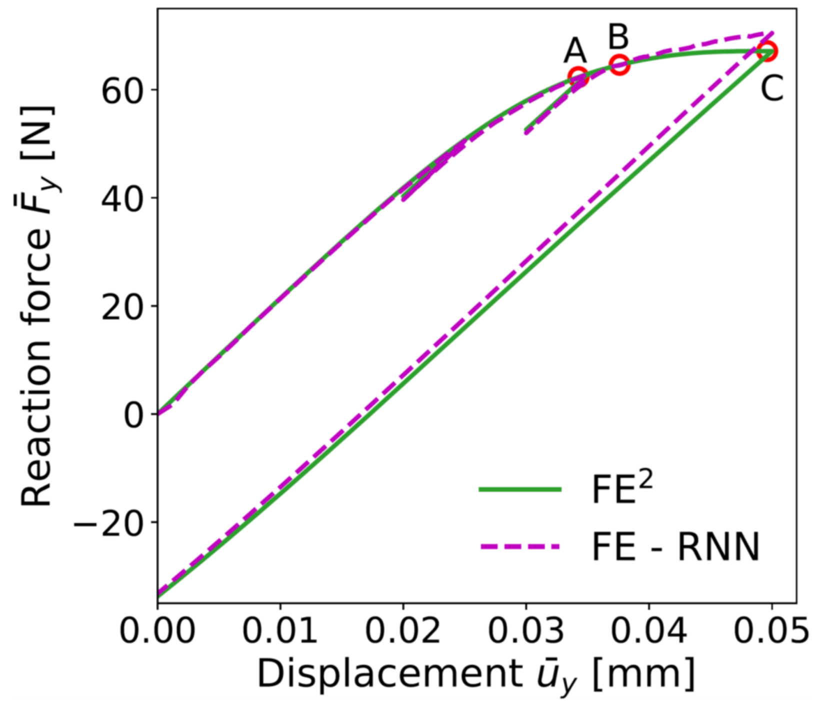


ANN as a mesoscale surrogate model

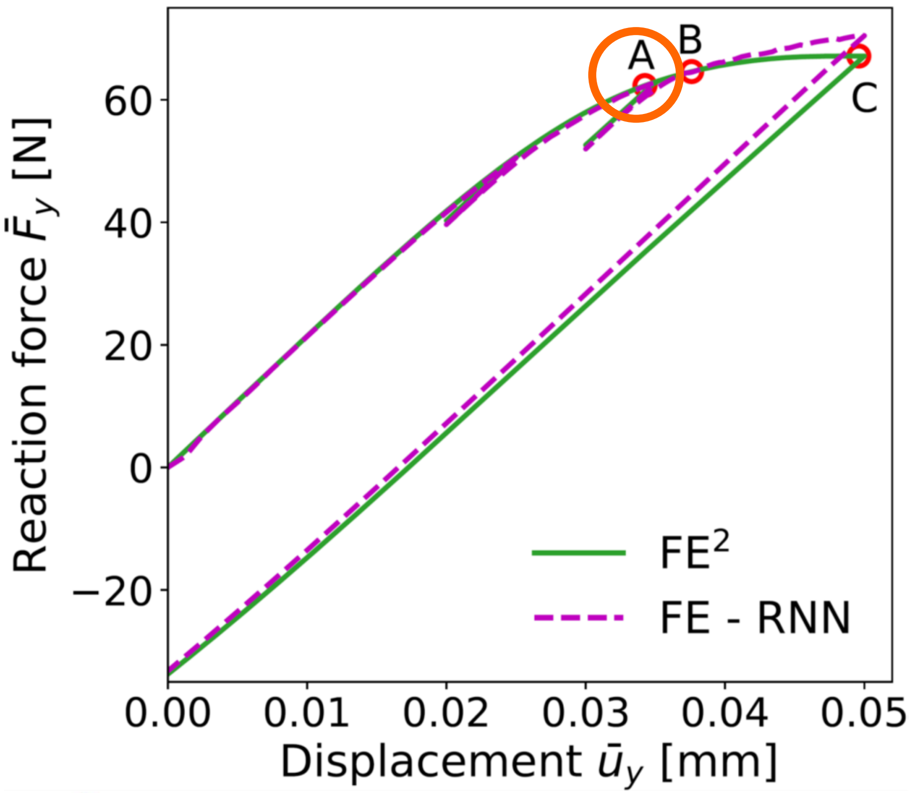
- Multiscale simulation
  - Elasto-plastic composite RVE
  - Comparison FE^2 vs. RNN-surrogate
  - Training data
    - Bounded at 10% deformation

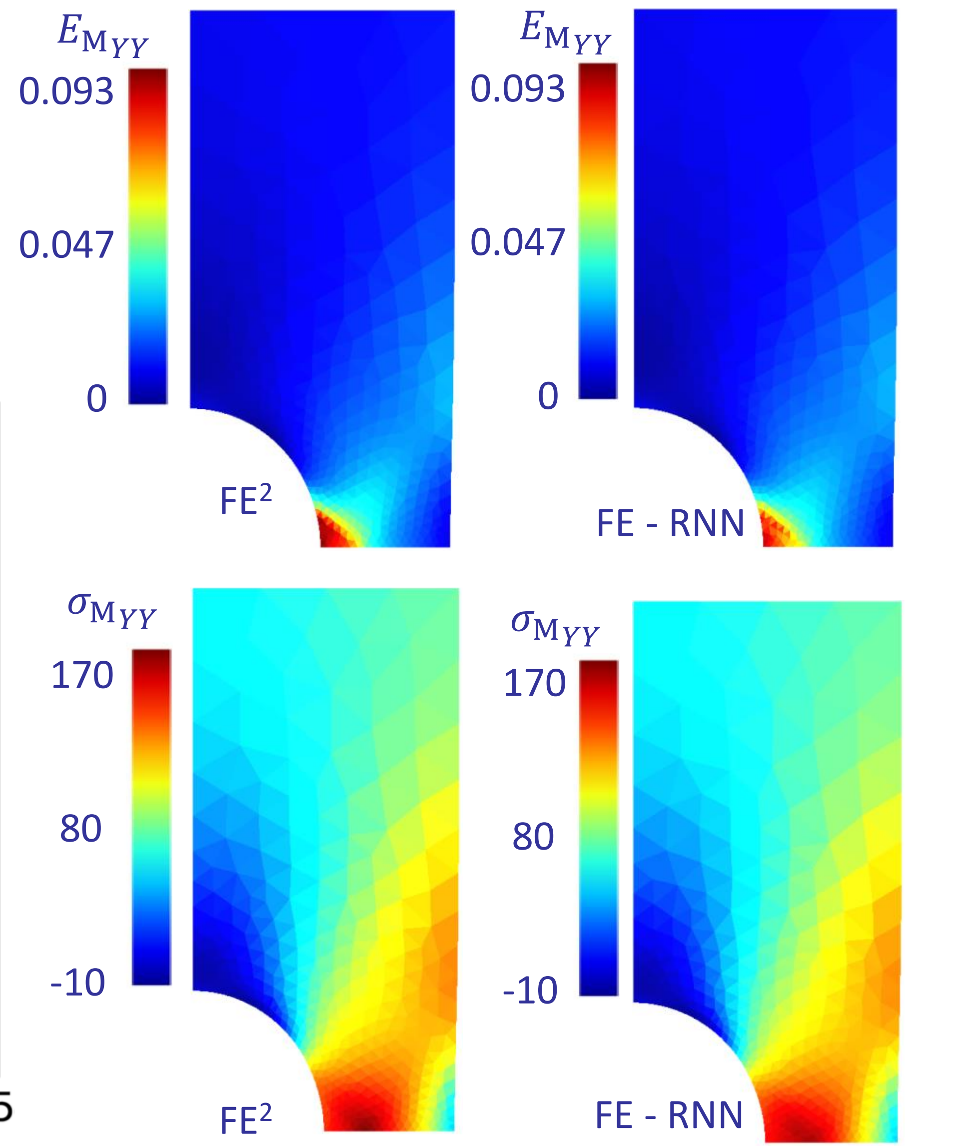
Off-line	FE^2	FE-RNN
Data generation	-	9000 x 2 h-cpu
Training	-	3 day-cpu
On-line	**FE^2**	**FE-RNN**
Simulation	18000 h-cpu	0.5 h-cpu





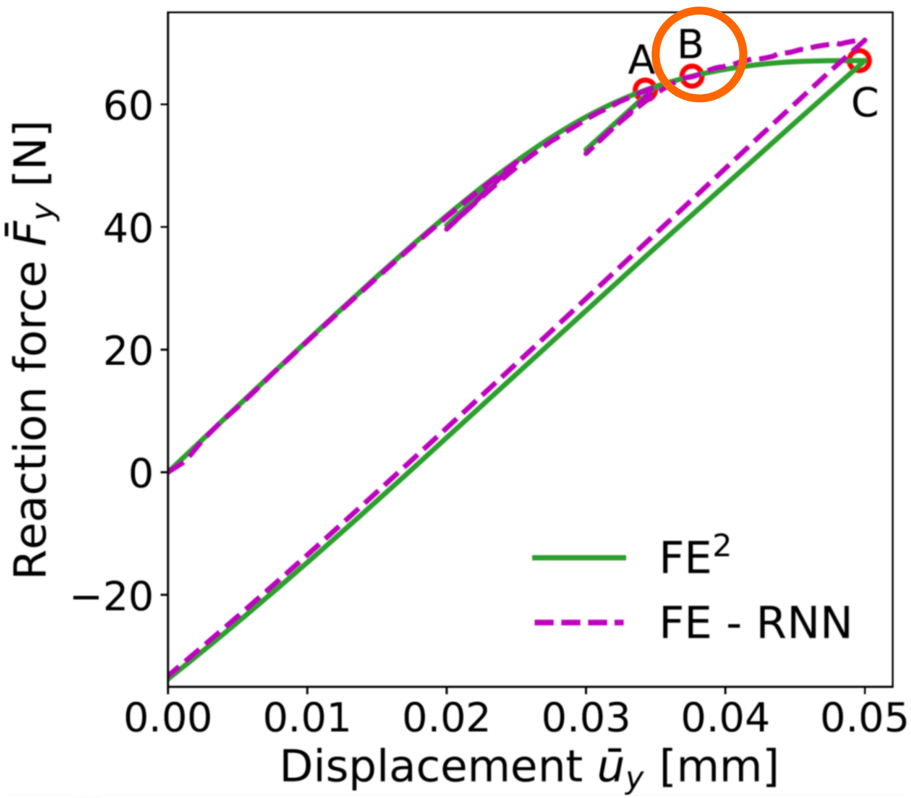
- Multiscale simulation
  - Stress-strain distribution at point A
  - Strain within the 10% training range

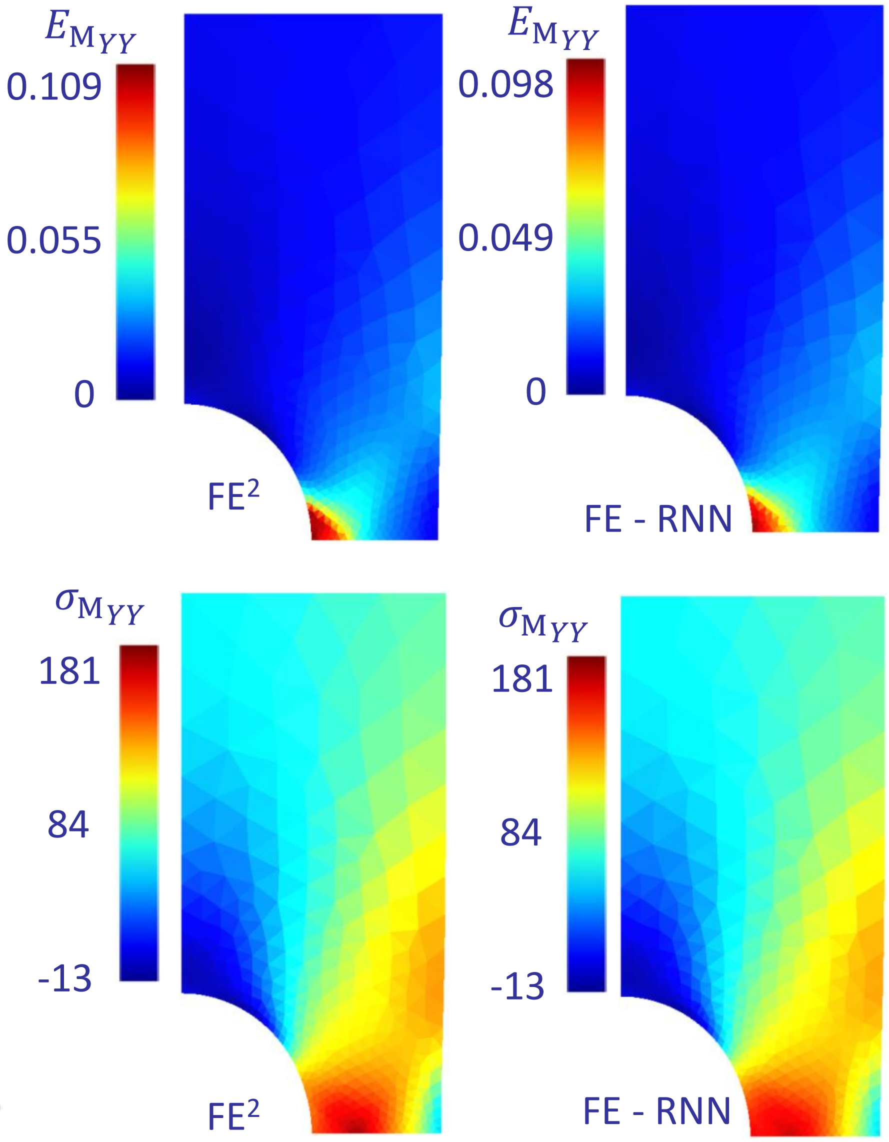




ANN as a mesoscale surrogate model

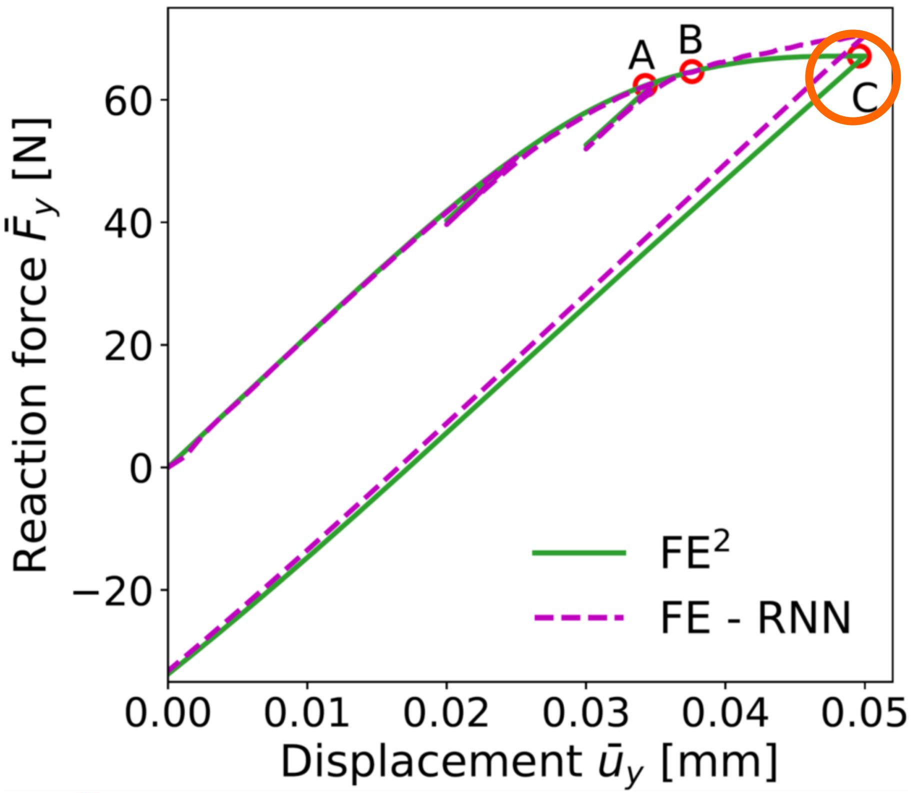
- Multiscale simulation
  - Stress-strain distribution at point B
  - Strain just at 10% training range

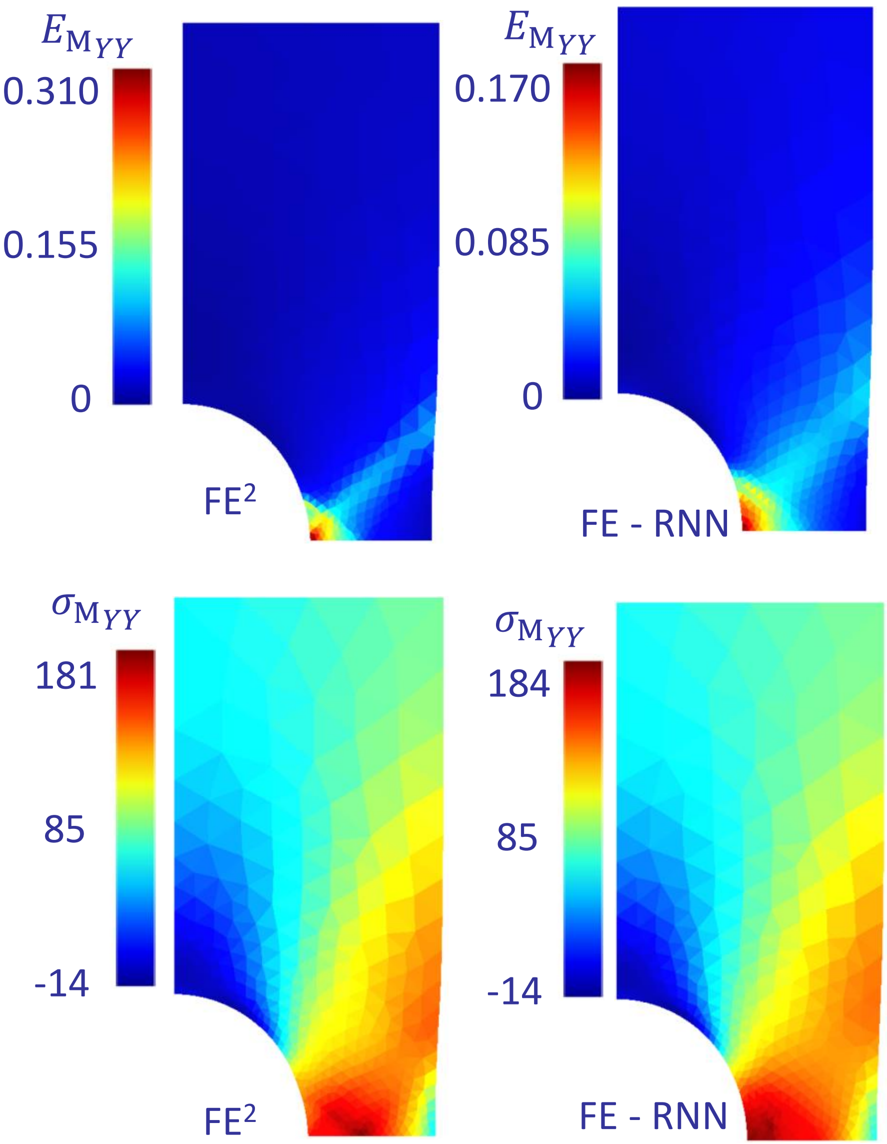




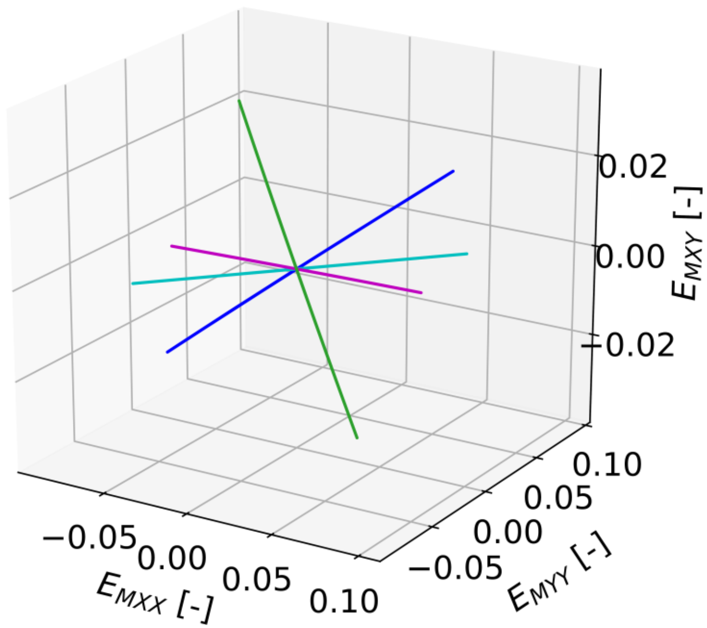
ANN as a mesoscale surrogate model

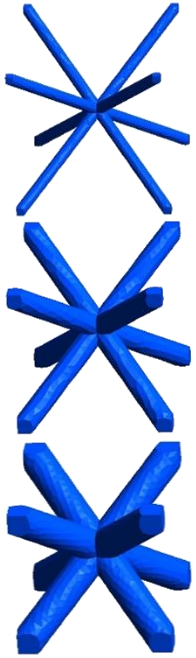
- Multiscale simulation
  - Stress-strain distribution at point C
  - Strain out of 10% training range





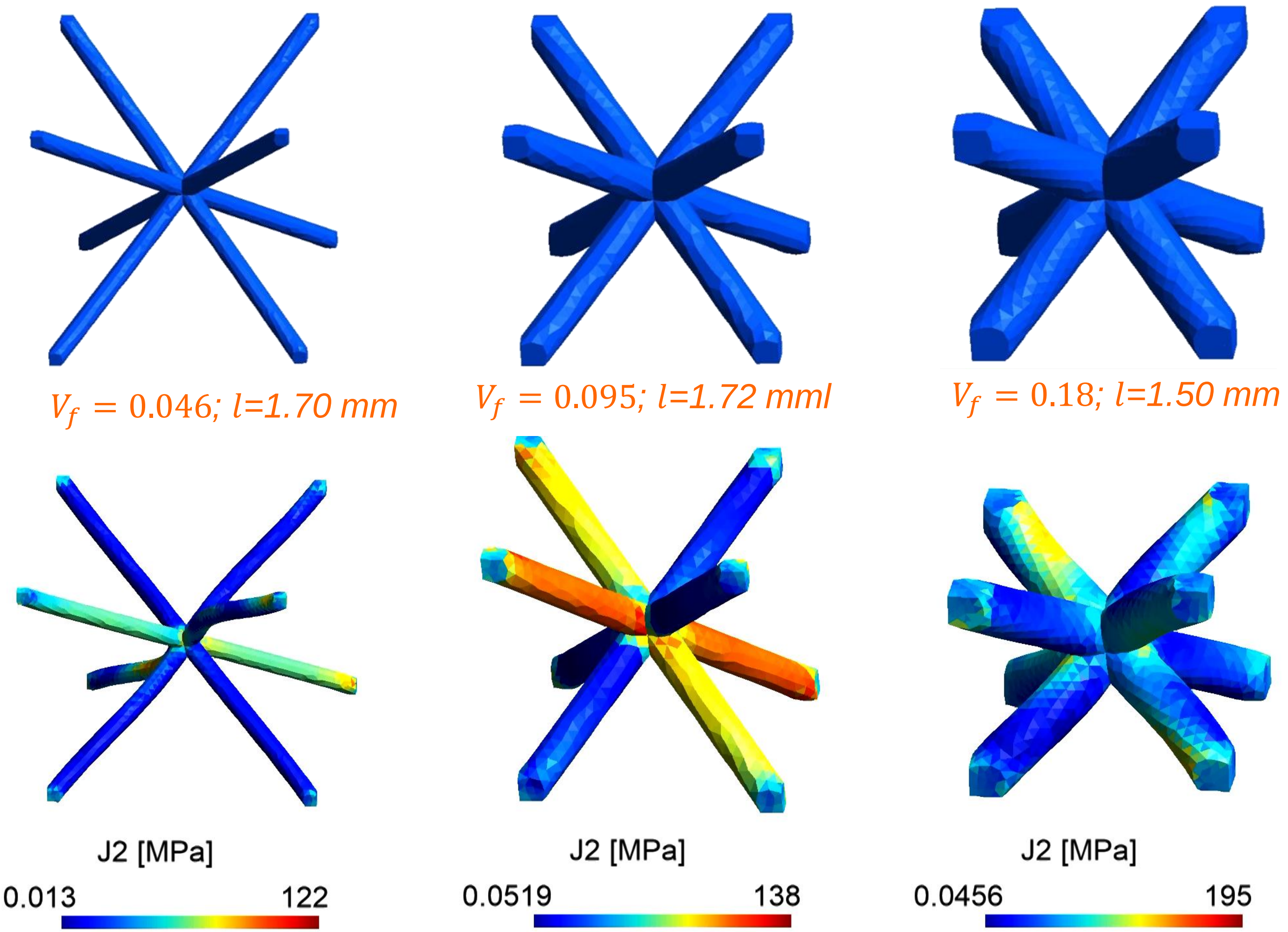
- Input / output definition
  - Input:
    - Strain (history): $\mathbf{F}_\mathrm{M}$
    - Geometry/material parameters: $\boldsymbol{\varphi}_\mathrm{m}$
  - Output:
    - Stress (history): $\mathbf{P}_\mathrm{M}$





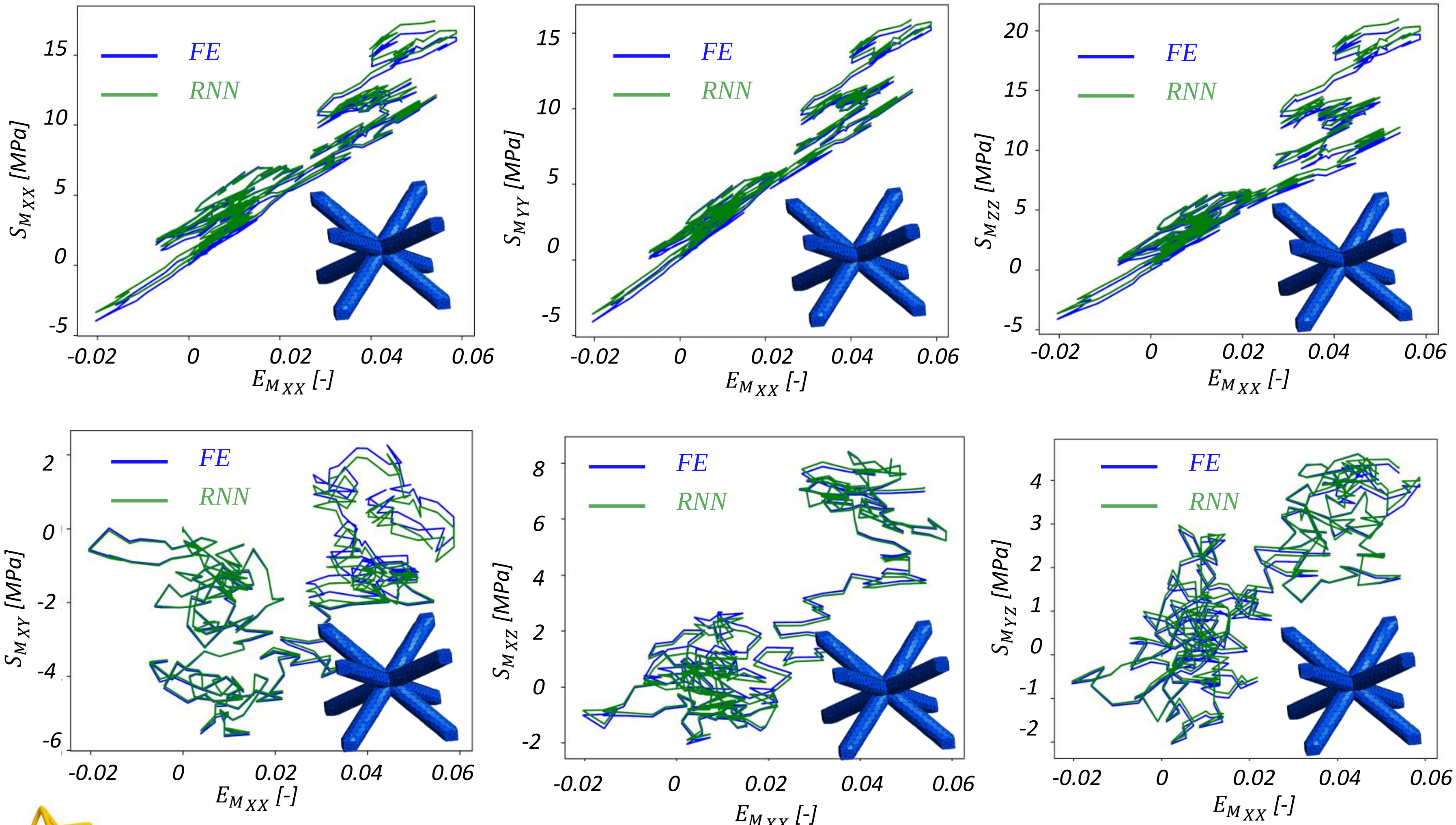
- Methodology
  - Address problem of geometry/material effect
    - Octet cells
    - Elastic material at first

- Octet cell
  - Generalised IMDEA script to generate random cells and random loading paths



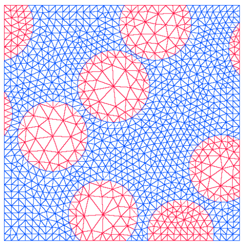
Geometrical parameters effect

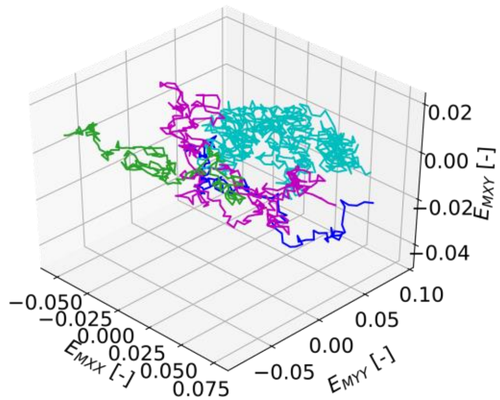
- Octet cell
  - Test on new random cell/path

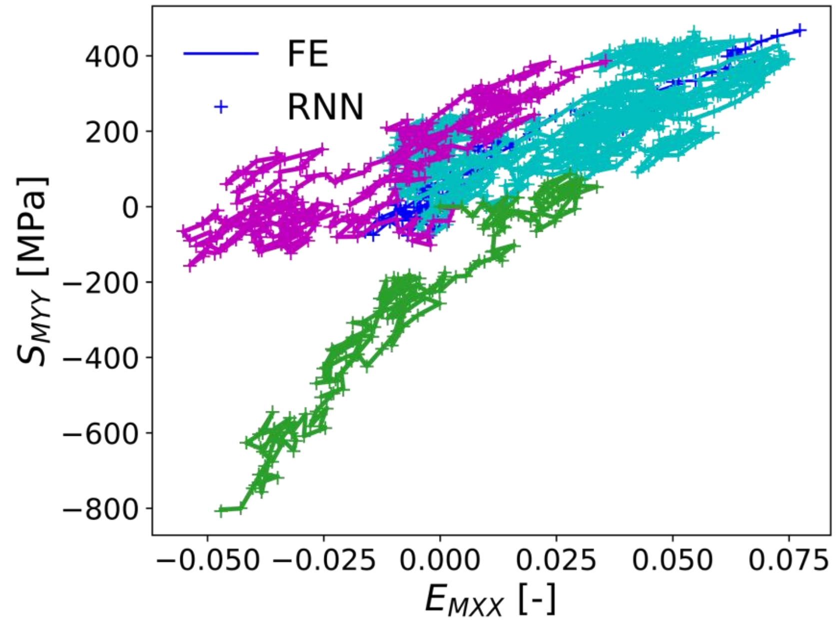


Localisation step

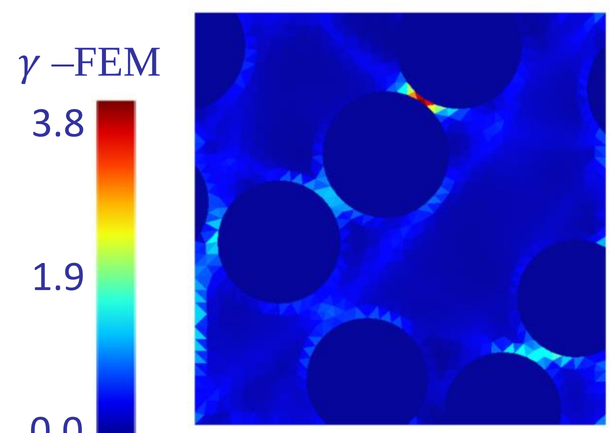
- Only homogenised output is predicted
  - On random walk





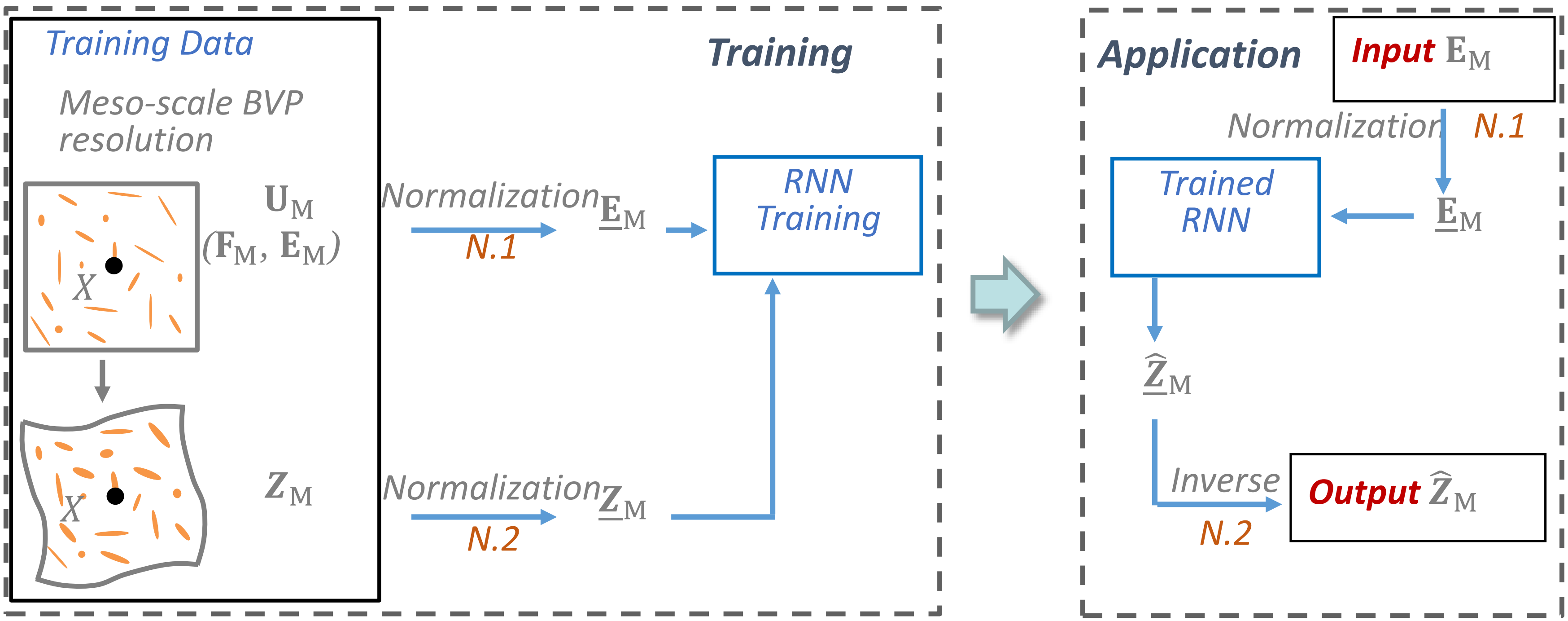


- Quid of local fields?
  - This is an advantage of multiscale methods
  - Useful to predict failure, fatigue etc.
  - Can we get it back at low cost?



Localisation step

- Also build a surrogate model of the internal variables

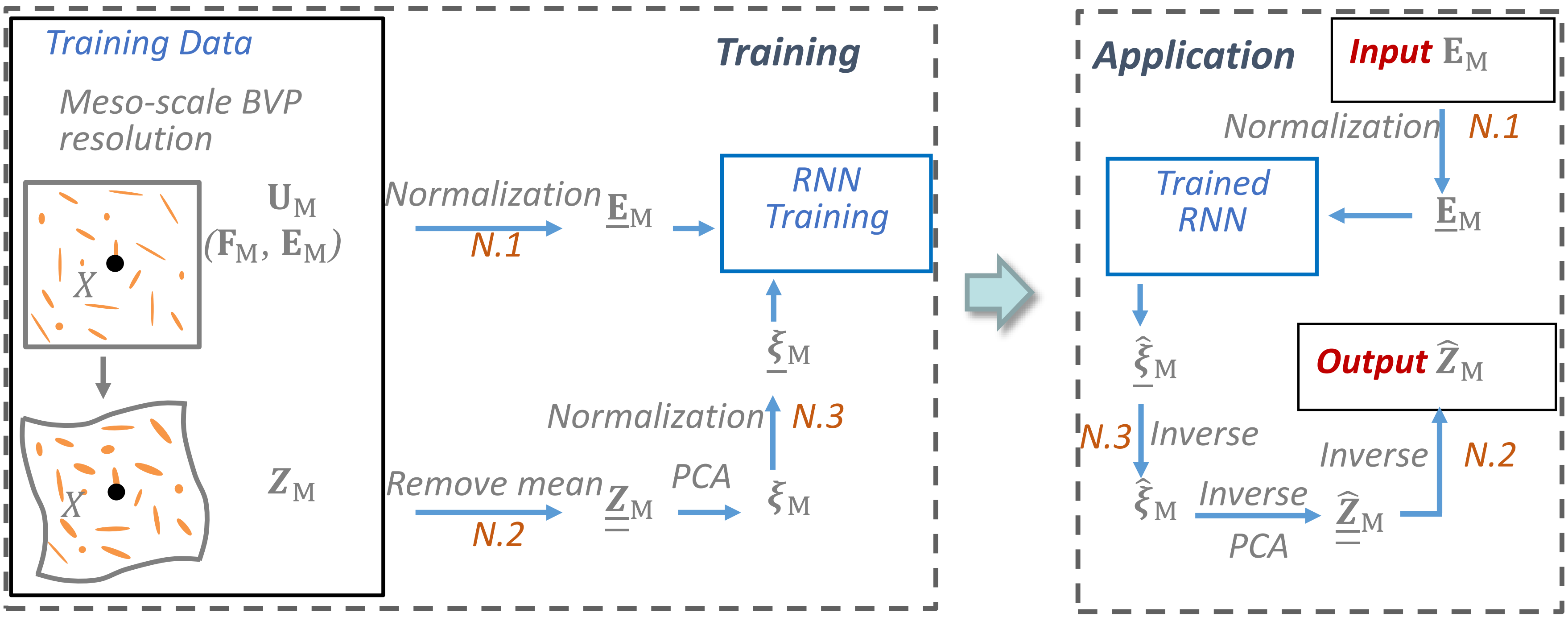


  - Problem: The size of $\underline{\boldsymbol{Z}}_\mathrm{M}$ is large
    - $\underline{\boldsymbol{Z}}_\mathrm{M}$ of size d the number of Gauss points of the RVE × internal variables by Gauss point

⇨ overwhelming cost

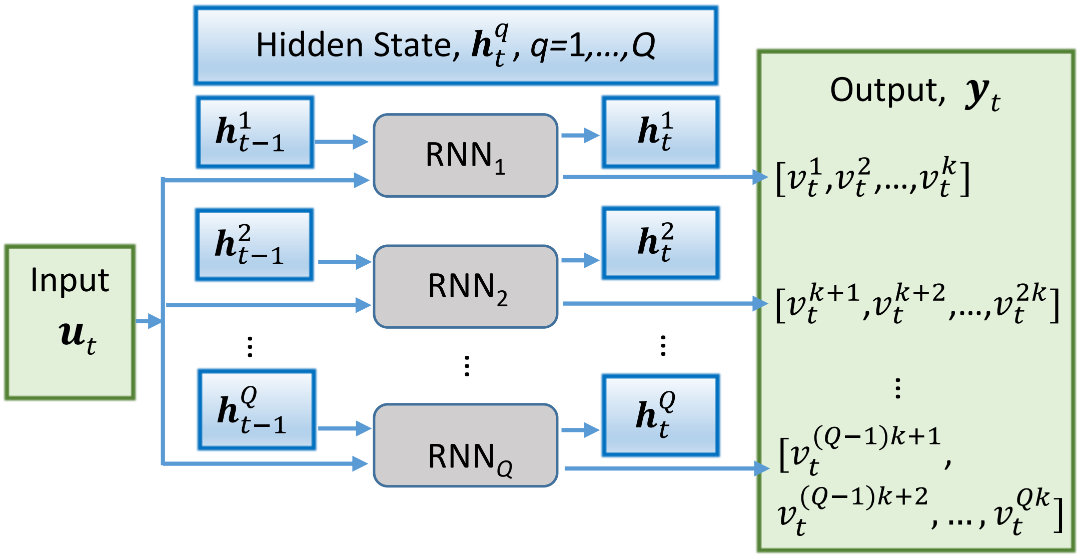
Localisation step

- Optimise the method: reduce the size of the internal variables



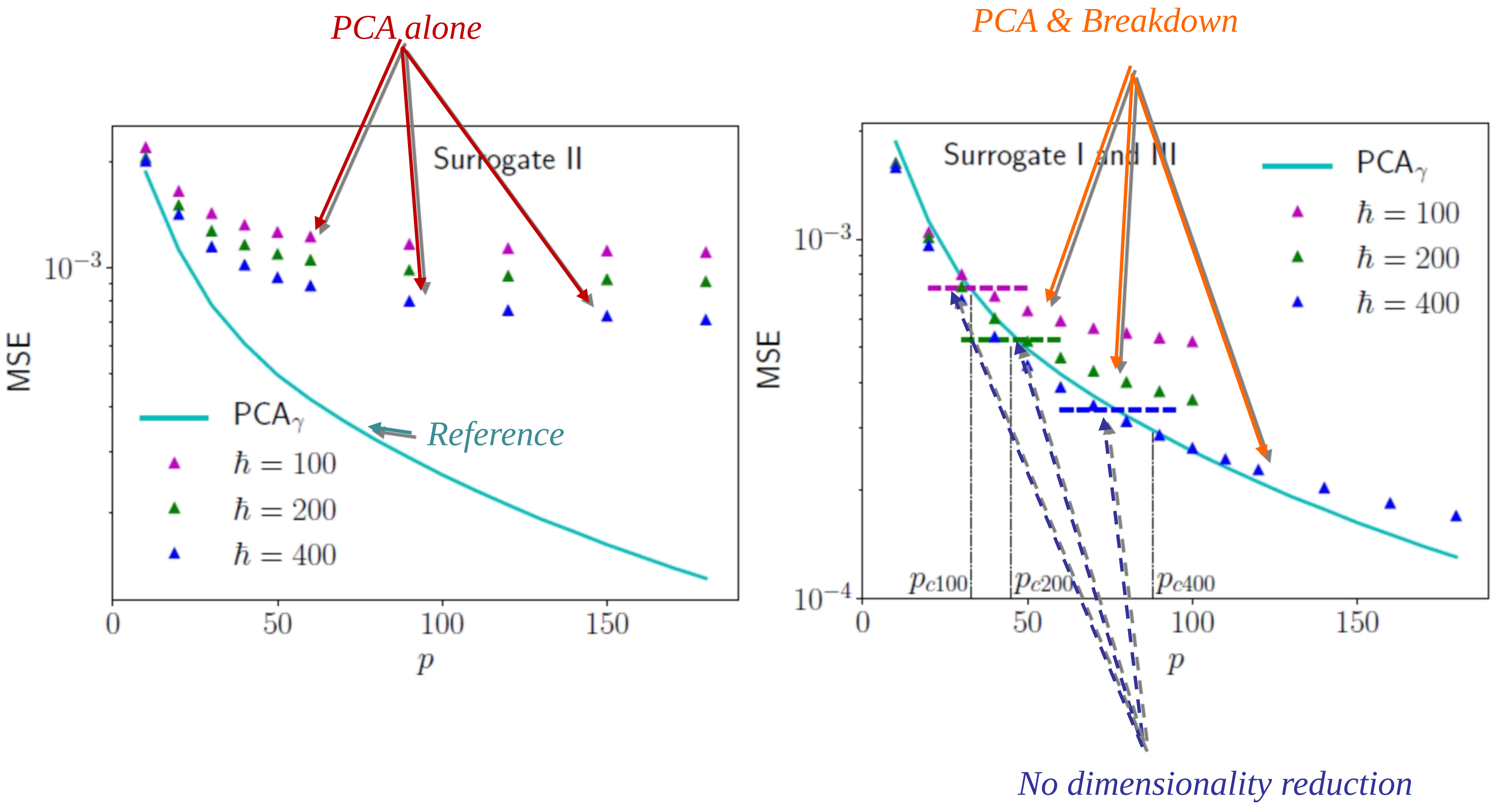
  - Principal Component Analysis (PCA) applied on $\boldsymbol{Z}_{\mathrm{M}}$ to reduce the output of RNN
    - Construct matrix $\mathbf{Z_M} = \left[\underline{\underline{Z}}_{\mathrm{M}_1}\ \underline{\underline{Z}}_{\mathrm{M}_2}\ \dots \underline{\underline{Z}}_{\mathrm{M}_n}\right]_{d\times n}$ from n observations (1% from all data)
    - Extract n ordered eigenvalues Λ_i and eigen vector $\underline{v}_i$ of $\mathbf{Z}_{\mathrm{M}}^T\mathbf{Z}_{\mathrm{M}}$
    - Build reduced basis $\mathbf{V} = \left[\ \underline{v}_1\ \underline{v}_2\ \ \dots \underline{v}_p\ \right]_{d\times p}$ and reduced data $\boldsymbol{\xi}_{\mathrm{M}} = \mathbf{V}^T\underline{\underline{\boldsymbol{Z}}}_{\mathrm{M}}$ of size $p < d$
    - Reconstruction $\underline{\underline{\hat{\boldsymbol{Z}}}}_{\mathrm{M}} = \mathbf{V}\boldsymbol{\xi}_{\mathrm{M}}$
    - But not enough

- Dimensionality reduction & break down

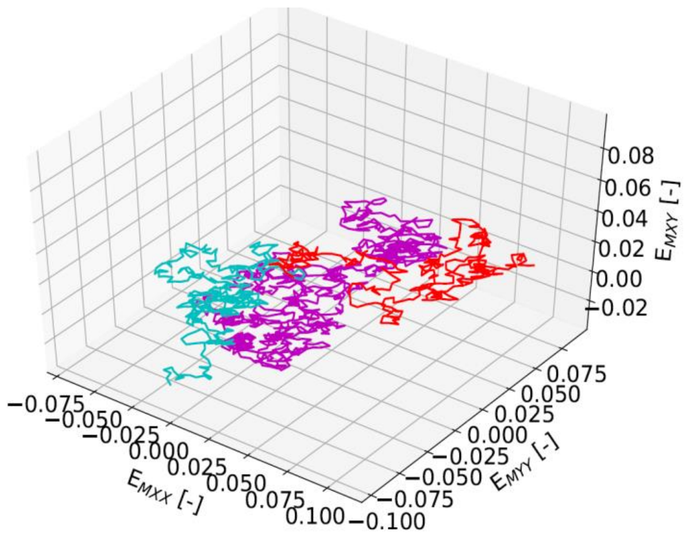


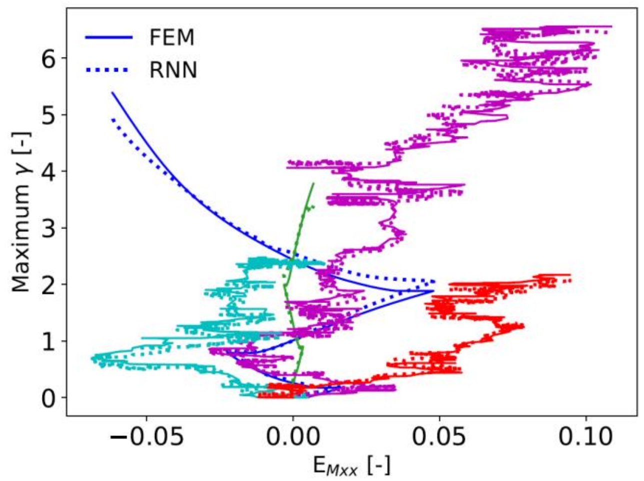
- To further reduce the output dimension of RNN
  - The surrogate modelling is carried out by a few small RNNs, instead of one big RNN
  - The high dimension output is divided into Q groups, and each RNN is used to reproduce only a part of output
- PCA reduces $\boldsymbol{Z}_{\mathrm{M}}$ to 180 outputs and we use Q=6

- Effect of dimensionality reduction and number of hidden variables

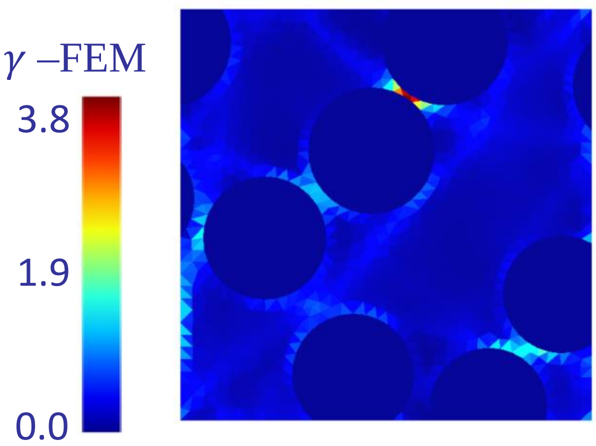


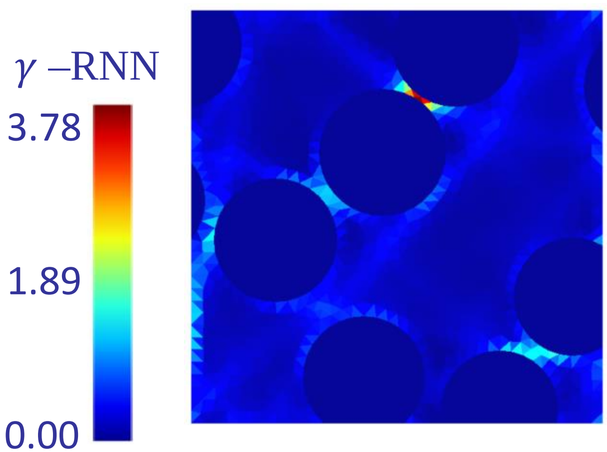
- Evaluation of equivalent plastic strain γ: Random loading (testing data)





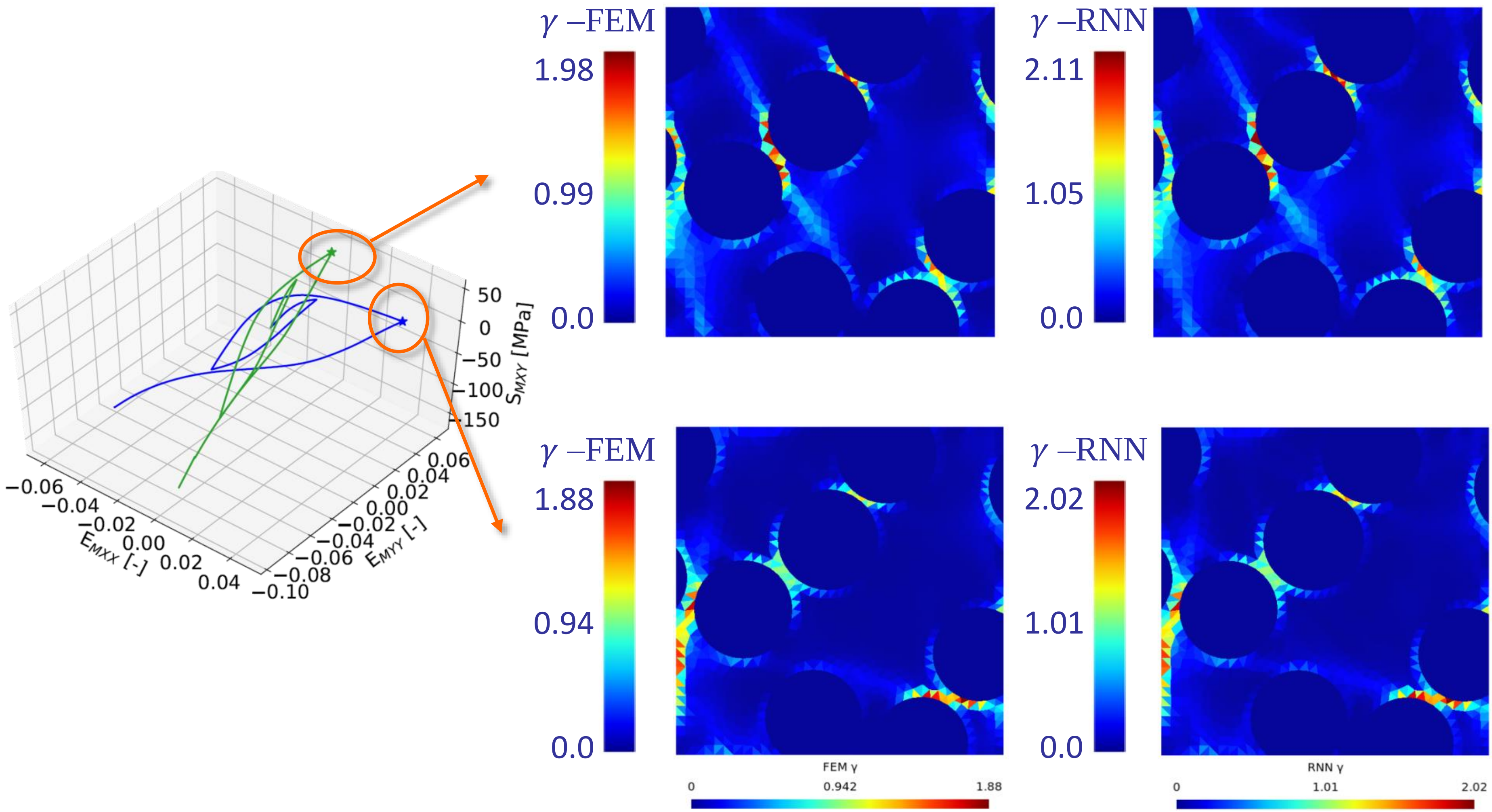
Purple loading – step 500





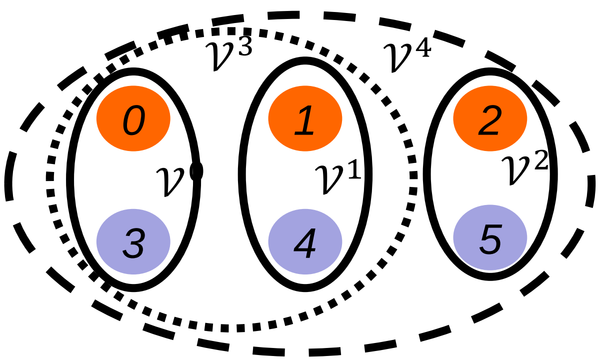
Localisation step

- Evaluation of equivalent plastic strain γ: Cyclic loading (testing data)



Perspective

- Promising methodology
  - For 3D problems, complex behaviours etc. requires more random loading paths
    - Order reduction might become mandatory to generate the synthetic database
  - New interesting approach: Deep Material Network
    - Original paper by Z. Liu, C. Wu, M. Koishi, Comp. Meth. in Appl. Mech. and Engng. 2019
    - Formalism rewritten by V.D Nguyen & L. Noels, Comp. Meth. in Appl. Mech. Engng. 2022



Strain averaging: $\bar{\boldsymbol{\varepsilon}} + \sum_j \alpha^{i,j} \boldsymbol{a}^j \otimes \boldsymbol{n}^j = \boldsymbol{\varepsilon}^i$ *for node i*

Hill-Mandel condition: $\left(\sum_{i \in \mathcal{V}^j} W^i \boldsymbol{\sigma}^i \alpha^{i,j} \right) \cdot \boldsymbol{n}^j = 0$ *for interaction j*

Boundary conditions: $\sum_{i \in \mathcal{V}^j} W^i \alpha^{i,j} = 0$ *for interaction j*

Constitutive behaviours: $\boldsymbol{\sigma}^p(t) = \boldsymbol{\sigma}^p\left(\boldsymbol{\varepsilon}^i(t), z(\tau \le t)\right)$ *for material p*

Offline stage: system parameters $\alpha^{i,j}, W^i, \boldsymbol{n}^j$*, evaluated by data-driven approaches*

Online stage: system unknowns $\boldsymbol{a}^j$ *evaluated by Newton-Raphson iterations*

References

- More on

  - www.moammm.eu
  - L. Wu, V. D. Nguyen, N. G. Kilingar, and L. Noels. "A recurrent neural network-accelerated multi-scale model for elasto-plastic heterogeneous materials subjected to random cyclic and non-proportional loading paths." Computer Methods in Applied Mechanics and Engineering 369 (September 1, 2020): 113234, http://dx.doi.org/10.1016/j.cma.2020.113234
  - Data on https://dx.doi.org/10.5281/zenodo.3902663
  - L. Wu, L. Noels. "Recurrent Neural Networks (RNNs) with dimensionality reduction and break down in computational mechanics; application to multi-scale localization step" Computer Methods in Applied Mechanics and Engineering (under revision)